

CLAIM SET AS AMENDED

1. (Currently Amended) A method of implementing the a multi-sectional encoding indexing method, ~~whether covering~~ comprising the steps of:

~~Step 1: An providing an m sectional encoding structure which is comprised of~~
including encoding structure values \bar{A}_i'' ; $i=0,1,\dots,m-1$, with each of the encoding
structure values \bar{A}_i'' has having at most 2^{k_i} varieties;

~~Step 2: transferring the encoding structure values \bar{A}_i'' can be transferred into n~~
encoding structure values \bar{A}_i'' . Upon transference, whereupon the encoding structure
values \bar{A}_i'' will have k_i bits varied, and thus the summation of the k_i bits associated
with the encoding structure values is equal to n , ($\sum_{i=0}^{m-1} k_i = n$), wherein n stands for the
an index bit width of the Indexing Method, multi-sectional encoding indexing
method;

~~Step 3: adjusting the encoding structure values \bar{A}_i'' operated into encoding~~
structure values \bar{A}_i' by using a bit-by-bit AND operation with $(2^n - 1)$ by AND bit by
bit, can be transferred to \bar{A}_i' of n the encoding structure values \bar{A}_i' having an n bit
width; and

determining whether or not the encoding structure values \bar{A}_i' are transferred
into $\bar{A}_0 \oplus \bar{A}_1 \oplus \dots \oplus \bar{A}_{m-1}$. Step 4: If and if the encoding structure values \bar{A}_i' is are

transferred into the $\bar{A}_0 \oplus \bar{A}_1 \oplus \dots \oplus \bar{A}_{m-1}$, then bits of \bar{A}_i , from $n - (\sum_{j=0}^{i-1} k_j + 1)$ to

$n - \sum_{j=0}^i k_j$, will become variable ones,

wherein ($\sum_{j=0}^{m-1} k_j = 0$ is assumed for short.) and the $\bar{A}_0 \oplus \bar{A}_1 \oplus \dots \oplus \bar{A}_{m-1}$ has a

feature characteristic that different output values can be produced as long as the values input inputted vary, i.e. having the and have a collision free effect. of "collision-free."

2. (Currently Amended) The method of implementing the a multi-sectional encoding indexing method, according to the claim 1, ~~two methods are presented and can be~~ wherein the transferring step is applied single individually or mingled mixed, as step 2 required. The purpose of both methods is to minimize and in both either cases, the k_i are minimized, every with a view to thereby shortening the index n bit width, in which, $\sum_{i=0}^{m-1} k_i = n$.

3. (Currently Amended) The method of implementing the a multi-sectional encoding indexing method, according to the claim 1, ~~the first method is to observe~~ wherein the transferring step includes observing each of the encoding structure values \bar{A}_i , and if the variable bits are not at the a right side, adjust them adjusting the bits.

4. (Currently Amended) The method of implementing ~~the a~~ multi-sectional encoding indexing method; according to the claim 1, the ~~second method can be~~ transferring step is applied under ~~the following~~ conditions when:

$\min(\bar{A}_i''') \leq \bar{A}_i''' \leq \max(\bar{A}_i''')$, and k_i ~~then satisfy~~ satisfies the formula:

$$2^{k_i-1} < (\max(\bar{A}_i''') - \min(\bar{A}_i''') + 1) \leq 2^{k_i},$$

~~add or subtract~~ performing the step of adding or subtracting a certain value to or from the encoding structure value, \bar{A}_i''' , thereby converting it into whose the k_i bits that are variable, wherein. ~~A particular~~ the certain value for subtraction is $\min(\bar{A}_i''')$.

5. (Currently Amended) An implementation method of ~~the a~~ reverse compensation indexing method, ~~whether covering~~ comprising the steps of:

~~Step 1: The providing a two-sectional encoding structure is comprised of~~
including encoding structure values \bar{A}_i''' ~~$i=0,1$~~ , where $i = 0, 1$, and the value of each
 encoding structure is $\min(\bar{A}_i''') \leq \bar{A}_i''' \leq \max(\bar{A}_i''')$ wherein
 k_i ~~in which~~ satisfy satisfies the following formula:

$$2^{k_i-1} < (\max(\bar{A}_i''') - \min(\bar{A}_i''') + 1) \leq 2^{k_i};$$

adding or subtracting a certain value to or from the encoding structure values
 \bar{A}_i''' , ~~Step 2: After and after~~ adding or subtracting a certain value to or from the
encoding structure values \bar{A}_i''' , k_0 bits of an encoding structure values \bar{A}_0''' and k_1 bits

of the encoding structure values \bar{A}_1 may be change, and wherein $\sum_{i=0}^1 k_i = n$, in which

n is the a width of the index bits;

Step 3: adjusting the encoding structure values \bar{A}_i , operated into encoding structure values \bar{A}_i by using a bit-by-bit AND operation with $(2^n - 1)$ by AND bit by bit, is transferred to the encoding structure values \bar{A}_i of having n bits;

Step 4: obtaining \bar{A}_0 by reversing the bits of \bar{A}_0 ; and

operating on \bar{A}_0 , obtained by reversing the bit locations of \bar{A}_0 , is operated with $\bar{A}_1 (= \bar{A}_1)$ by exclusive-OR, i.e., $(\bar{A}_0 \oplus \bar{A}_1)$, thereby producing; The feature is that the output values produced by the indexing method will be which are different if input of different values are inputted, i.e. having the and have a collision free effect. of "collision free."

6. (Currently Amended) The method of implementing the multi-sectional encoding a reverse compensation indexing method; according to the claim 5, the outstanding property of the reverse compensation indexing method to be illustrated: wherein if $\min(VPI) = 0$ and $\min(VCI) = 0$, then this a single circuit can be applied in

$$\begin{aligned} \max(VPI) &= 2^{k_0} - 1 \\ \max(VCI) &= 2^{n-k_0} - 1, k_0 = 0, 1, \dots, n \end{aligned}$$

up to (n+1) occasions, all of which have the collision free effect. ~~of "collision free."~~

7. (Currently Amended) A method ~~that corresponds for implementing~~ multi-sectional encoding structures corresponding to a single indexing table ~~is presented. The in an Asynchronous Transfer Mode receiving unit of the ATM can support which supports different VPI/VCI Virtual Path Identifier/Virtual Connection Identifier (VPI/VCI) connection amounts by relatively entries, comprising the step of:~~

~~adding to or subtracting from the Indexing Table Memory an indexing table memory equipped in the said receiving unit, in coordination by coordinating with the TBWA a Table Bit Width Adjuster (TBWA) circuit; and~~

~~The changing an input inputted exclusive-OR configuration of the VPI and the VCI may be changed according to the a bit width of the indexing table Memory memory.~~

8. (Currently Amended) A structure ~~is presented that corresponds for implementing~~ multi-sectional encoding structures to shared memories in a shared memory structure of an Asynchronous Transfer Mode (ATM) switch, comprising the step of:

providing a Table Bit Width Adjuster (TBWA) circuit with multiple input ports r, and an indexing table memory, the indexing table memory have sections ~~The Indexing Table Memory of for e entries, from Entry 0 to Entry e-1, i.e. wherein Entry 0, Entry 1, ...to Entry~~

~~e-1, and~~ $\sum_{i=0}^{r-1} 2^{TBW_i} = e,$

wherein then the Base Pointer a base pointer (BP) of the an input unit 0 is BP_0 , and
input units these of the base pointer pointers for other $r-1$ input ports is figured out
determined by the Table Bit Width (TBW) and Base Pointer (BP) of the previous unit, Its
using a formula is:

$$BP_i = BP_{i-1} + W * 2^{TBW_{i-1}}, i = 1, \dots, r-1$$

wherein The purpose of calculating every BP base pointer is calculated in order to
orderly and compatibly allocate every section of the indexing table memory used by different
input ports. Such kind of whereby the shared memory structure can more effectively adjust
the adjusts a connection amount of every input port, thereby with limited minimizing an
amount of indexing table memory which is required.

9. (Currently Amended) An applied A method of for implementing multi-sectional
encoding structures which divided into sets of two sections, wherein and every set is
regarded as a basic unit for implementing the a reverse compensation indexing method.
Therefore, this kind of multi-sectional encoding structures can have the same flexibility
necessary for application of the reverse compensation indexing method.